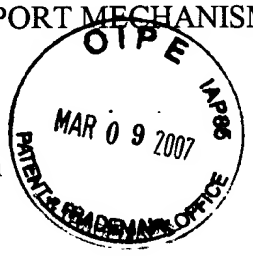
IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Applicant: Baber Amin et al.

Title: COMPUTER NETWORK HAVING A SECURITY LAYER INTERFACE INDEPENDENT OF THE APPLICATION TRANSPORT MECHANISM



Docket No.: 1565.023US1
Filed: July 20, 2000
Examiner: Andrew L. Nalven

Serial No.: 09/620,176
Due Date: March 6, 2007
Group Art Unit: 2134

MS Appeal Brief - Patents
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

We are transmitting herewith the following attached items (as indicated with an "X"):

X Appeal Brief Under 37 CFR 41.37 (24 pgs.) including authorization to charge Deposit Account 19-0743 in the amount of $500.00 to cover the Appeal Fee.
X Return postcard.

If not provided for in a separate paper filed herewith, Please consider this a PETITION FOR EXTENSION OF TIME for sufficient number of months to enter these papers and please charge any additional fees or credit overpayment to Deposit Account No. 19-0743.

SCHWEGMAN, LUNDBERG, WOESSNER & KLUTH, P.A.
Customer Number 21186

By: /Joseph P. Mehrle/
Atty: Joseph P. Mehrle
Reg. No. 45,535

CERTIFICATE UNDER 37 CFR 1.8: The undersigned hereby certifies that this correspondence is being deposited with the United States Postal Service with sufficient postage as first class mail, in an envelope addressed to: MS Appeal Brief - Patents, Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450, on this 6 day of March, 2007.

Peter Rebuffoni
Name

Peter Rebuffoni
Signature

PATENT

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Appellants: Baber Amin et al.

Examiner: Andrew L. Nalven

Serial No.: 09/620,176

Group Art Unit: 2134

Filed: July 20, 2000

Docket: 1565.023US1

Title: COMPUTER NETWORK HAVING A SECURITY LAYER INTERFACE INDEPENDENT OF THE APPLICATION TRANSPORT MECHANISM

APPEAL BRIEF UNDER 37 CFR § 41.37

Mail Stop Appeal Brief- Patents
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

The Appeal Brief is presented in response to the Notice of Panel Decision from Pre-Appeal Brief Review mailed on February 6, 2007 and further in support of the Notice of Appeal to the Board of Patent Appeals and Interferences, filed on January 3, 2007, from the Final Rejection of claims 1-20 of the above-identified application, as set forth in the Final Office Action mailed on November 2, 2006.

The Commissioner of Patents and Trademarks is hereby authorized to charge Deposit Account No. 19-0743 in the amount of $500.00 which represents the requisite fee set forth in 37 C.F.R. § 41.20(b)(2). The Appellants respectfully request consideration and reversal of the Examiner's rejections of pending claims.

APPEAL BRIEF UNDER 37 C.F.R. § 41.37

TABLE OF CONTENTS

1. REAL PARTY IN INTEREST

The real party in interest of the above-captioned patent application is the assignee, NOVELL, INC., as evidenced by the assignment from the inventors recorded July 20, 2000 at Reel 011038, Frame 0243.

2. RELATED APPEALS AND INTERFERENCES

There are no other appeals or interferences known to Appellants that will have a bearing on the Board's decision in the present appeal.

3. STATUS OF THE CLAIMS

The present application was filed on July 20, 2000 with claims 1-20. A non-final Office Action was mailed May 16, 2006. A Final Office Action (hereinafter "the Final Office Action") was mailed November 2, 2006. Claims 1-20 stand twice rejected, remain pending, and are the subject of the present Appeal.

4. STATUS OF AMENDMENTS

No amendments have been made subsequent to the Final Office Action mailed November 2, 2006.

5. SUMMARY OF CLAIMED SUBJECT MATTER

Some aspects of the present inventive subject matter include, but are not limited to, a computer network having security layer interface independent of the application transport mechanism.

INDEPENDENT CLAIM 1

1. A method of providing transport-independent secure communications in a computer network, comprising the steps of: *[FIGS. 5-8; specification pages 9-14]*

directly receiving application data, from an application, at an upper connection layer of a transport protocol stack, wherein the application data is received from the application using a connection specific application programming interface (API) desired for communication by the application and which is not associated with security; *[FIG. 5 reference numerals 200 and 202; FIG. 7 reference numerals 200 and 500; FIG. 8 reference numeral 200 and 500; specification page 3 first paragraph under the Brief Summary of the Invention; page 7 second full paragraph; page 9 last full paragraph and continuing to page 10 first full paragraph; page 13 section entitled Application Data Transfer and continuing through page 14]*

passing the application data from the upper connection layer to a security layer from within the transport protocol stack and unbeknownst to the application; *[specification page 7 last full paragraph; FIG. 5 reference numeral 202 and 502; FIG. 6 reference numerals 500 and 502; FIG. 7 reference numerals 500 and 502; FIG. 8 reference numeral 500 and 502; specification page 12 and page 13 section entitled Application Data Transfer and continuing through page 14]*

encrypting the application data within the security layer; *[FIG. 5 reference numeral 206; FIG. 6 reference numeral 206; FIG. 7 reference numeral 206; FIG. 8 reference numeral 206; specification page 7 last paragraph; page 8 first full paragraph; page 12 penultimate paragraph; page 13 paragraph beginning with step 4; page 16 paragraph beginning with step 6]*

passing the encrypted application data from the security layer to a lower connection layer of the transport protocol stack; and *[FIG. 7 step 6; FIG. 7 steps 6, 2, and 2a; specification page*

3 first paragraph under the Brief Summary of the Invention; page 14 paragraph beginning with step 6; page 26 last full paragraph]

sending the encrypted application data from the lower connection layer out a network connection; *[FIGS. 6-8 Data out on wire label; specification page 3 first paragraph under the Brief Summary of the Invention; original filed claim 1]*

wherein the application is not required to perform security handshakes in order to send encrypted application data over the network, the connection layers support at least one network transport protocol, and the security layer is not specific to that transport protocol. *[specification page 3 first paragraph under the Brief Summary of the Invention; page 3 last paragraph and continuing to the first paragraph on page 4; first full paragraph on page 4; last paragraph page 4 and continuing to the first paragraph on page 5; page 7 last paragraph; page 12 penultimate paragraph; page 13 paragraph beginning with step 4; page 26 last paragraph]*

INDEPENDENT CLAIM 7

7. A system for secure computer networking, comprising: *[FIGS. 5-8; specification pages 9-14]*

an application which is free of code for performing security procedure handshakes for secure network communications; *[FIGS 5, 7-8 reference numeral 200; specification page 4 first full paragraph; original filed claim 7; FIG. 4 reference numeral 200; specification page 6 third paragraph; page 7 first full paragraph; page 7 last paragraph; page 10 first full paragraph]*

at least one connection layer directly interfaced with the application, the connection layer comprising an upper connection layer associated with a transport protocol stack and a lower connection layer associated with the transport protocol stack, the connection layers comprising code for performing at least one network transport protocol; and *[FIGS. 5-8 reference numerals 202, 500, 206, 502; specification page 3; page 4 penultimate paragraph; page 5first incomplete paragraph; page 7; pages 10-11]*

a security layer callable from the connection layer rather than the application and wherein the security layer is unbeknownst to the application, the security layer comprising code for performing security procedure handshakes for secure network communications, the security layer also comprising code for encrypting and decrypting application data, and wherein the

application initially sends application data to the protocol stack of the upper connection layer directly using a desired application programming interface (API) associated with a connection mechanism that is not associated with security. *[FIGS. 5-8 reference numeral 206; specification page 3 first paragraph under the Brief Summary of the Invention; page 3 last paragraph and continuing to first paragraph page 4; page 4 first full paragraph; page 4 penultimate paragraph; page 5 first incomplete paragraph; page 7 first full paragraph; page 26 last full paragraph; original filed claims 1 and 7]*

INDEPENDENT CLAIM 16

16. A configured storage medium embodying data and instructions readable by a computer to perform a method of processing application data for secure network communications, the method comprising the computer-implemented steps of: *[specification page 4 last paragraph and continuing to the first paragraph on page 5; original filed claim 16; FIGS. 5-8]*

at a security layer, receiving a request from a lower connection layer of a transport protocol stack to establish a secure connection, wherein an application that utilizes the security layer is unaware of the security layer and its operations; [FIG. 5 reference numerals 202, 206, and 502; FIG. 6 step 2 reference numerals 502 and 206; FIG. 7 step 2 reference numerals 502 and 206; FIG. 8 step 2 reference numerals 502 and 206; specification pages 12-14]

in response, utilizing a means for establishing a connection to establish the requested connection; and *[specification page 5 first incomplete paragraph; original filed claim 16; FIG. 8 step 1a; pages 12-14]*

at the security layer, receiving encrypted application data from the lower connection layer, decrypting the application data, and passing the decrypted application data to an upper connection layer of the transport protocol stack; *[FIGS. 5-8 reference numeral 206; specification pages 19-21]*

whereby the application directly receives the decrypted application data without being required to perform security procedure handshakes for secure network communications and without being aware of security communications that occur via the security layer, and wherein the application receives the decrypted application data in a desired application programming interface (API) associated with a connection that the application originally used and that is not

associated with security. *[specification page 3 first paragraph under the Brief Summary of the Invention; page 3 last paragraph and continuing to the first paragraph on page 4; first full paragraph on page 4; last paragraph page 4 and continuing to the first paragraph on page 5; page 7 last paragraph; page 12 penultimate paragraph; page 13 paragraph beginning with step 4; page 26 last paragraph]*

This summary does not provide an exhaustive or exclusive view of the present subject matter, and Appellant refers to the appended claims and its legal equivalents for a complete statement of the invention.

6. GROUNDS OF REJECTION TO BE REVIEWED ON APPEAL

§102 Rejection of the Claims

Claims 1-2, 4, 6-9, 12, 14-18 and 20 were rejected under 35 U.S.C. § 102(e) for anticipation by Tumblin et al. (U.S. 6,490,679 – herein after Tumblin).

§103 Rejection of the Claims

Claims 3 and 10 were rejected under 35 U.S.C. § 103(a) as being unpatentable over Tumblin et al. in view of SSL-Talk List FAQ Secure Sockets Layer Discussion List FAQ v1.1.1 ("SSL-Talk List FAQ").

Claim 5 was rejected under 35 USC § 103(a) as being unpatentable over Tumblin et al. in view of Samar (U.S. 6,304,974 – hereinafter Samar).

Claims 11 and 19 were rejected under 35 USC § 103(a) as being unpatentable over Tumblin et al. in view of Novell NetWare Connection Enhanced NetWare 5 "What's Enhanced in NetWare 5."

Claim 13 was rejected under 35 USC § 103(a) as being unpatentable over Tumblin in view of Microsoft Security Advisor SSL Specific WSAIoctl Controls ("MS SSL Advisor").

7. ARGUMENT

A) The Applicable Law under 35 U.S.C. §102(e).

A claim is anticipated only if each and every element as set forth in the claim is found, either expressly or inherently described, in a single prior art reference. M.P.E.P § 2131. To anticipate a claim, a reference must disclose every element of the challenged claim and enable one skilled in the art to make the anticipating subject matter. PPG Industries, Inc. V. Guardian Industries Corp., 75 F.3d 1558, 37 USPQ2d 1618 (Fed. Cir. 1996). The identical invention must be shown in as complete detail as is contained in the claim. Richardson v. Suzuki Motor Co., 868 F.2d 1226, 1236, 9 USPQ2d 1913, 1920 (Fed. Cir. 1989). The elements must be arranged as required by the claims.

Appellant would like to respectfully reiterate that anticipation is only proper if "[t]he identical invention must be shown in as complete detail as contained in the . . . claim." *Richardson v. Suzuki Motor Co.*, 868 F.2d 1226, 1236, 9 USPQ 1913, 1920 (Fed. Cir. 1989).

B) The Applicable Law under 35 U.S.C. §103(a).

To sustain a rejection under 35 U.S.C. 103, references must be cited that teach or suggest all the claim elements. M.P.E.P. § 2142 (citing *In re Vaeck*, 947 F.2d 488, 20 USPQ2d 1438 (Fed. Cir. 1991)). In determining the differences between the prior art and the claims, the question under 35 U.S.C. 103 is not whether the differences themselves would have been obvious, but whether the claimed invention as a whole would have been obvious. *Stratoflex, Inc. v. Aeroquip Corp.*, 713 F.2d 1530, 218 USPQ 871 (Fed. Cir. 1983); *Schenck v. Nortron Corp.*, 713 F.2d 782, 218 USPQ 698 (Fed. Cir. 1983); *Interconnect Planning Corp. v. Feil*, 774 F.2d 1132, 1143, 227 USPQ 543, 551 (Fed. Cir. 1985); MPEP § 2141.02.

Further, the teaching or suggestion to make the claimed combination and the reasonable expectation of success must both be found in the prior art, not in Appellant's disclosure. *In re Vaeck*, 947 F.2d 488, 20 USPQ2d 1438 (Fed. Cir. 1991); MPEP § 2143. The Examiner must

avoid hindsight. *In re Bond*, 910 F.2d 831, 834, 15 USPQ2d 1566, 1568 (Fed. Cir. 1990). The Office Action must further provide specific, objective evidence of record for a finding of a suggestion or motivation to combine reference teachings and must explain the reasoning by which the evidence is deemed to support such a finding. *In re Sang Su Lee*, 277 F.3d 1338, 61 USPQ2d 1430 (Fed. Cir. 2002).

Appellants would further like to point out that the mere fact that references can be combined or modified does not render the resultant combination obvious unless the prior art recited also suggests in some manner the desirability of the proposed combination. *In re Mills*, 916 F.2d 680, 16 USPQ 2d 1430 (Fed. Cir. 1990). Appellants would also like to note that "rejections on obviousness grounds cannot be sustained by mere conclusory statements; instead, there must be some articulated reasoning with some rational underpinning to support the legal conclusion of obviousness. *See Lee*, 277 F.3d 1338, 1343-46 (Fed. Cir. 2002); *Rouffet*, 149 F.3d 1350, 1355-59 (Fed. Cir. 1998). This requirement is rooted in the Administrative Procedure Act, which ensures due process and non-arbitrary decision making, as it is in 35 U.S.C. § 103. See id, at 1344-45." *In re Kahn*, No. 04-1616 (Fed. Cir. March 22, 2006).

C) Discussion of the rejections of claims 1-2, 4, 6-9, 12, 14-18 and 20 under 35 U.S.C. § 102(e) as being anticipation by Tumblin.

Claims 1-2, 4, 6-9, 12, 14-18 and 20 were rejected under 35 U.S.C. § 102(e) for anticipation by Tumblin. Appellant respectfully traverses this rejection because Tumblin fails to teach or suggest an application that benefits from security and that is unaware of that security and is not pre-configured to support the security. The specific claim limitations that support this contention will be discussed in context herein and below. The thrust of the disagreement with the Examiner is in that the Examiner contends that Tumblin supports a teaching where an application that benefits from security is unaware of that security within Tumblin. The Appellant disagrees with the interpretation of Tumblin and with the application of Tumblin to the claim language and believes that there has been no proper prima facie showing of anticipation.

Specifically, pending independent claim 1 recites the following limitations: "directly receiving application data, from an application, at an upper connection layer . . . received from the application using a connection specific application programming interface . . . and which is

not associated with security; "passing the application data from the upper connection layer to a security layer . . . unbeknownst to the application;" and "the application is not required to perform security handshakes . . . and the security layer is not specific to that transport protocol." Independent claim 7 also includes limitations for "an application free of code for performing security handshakes" and a "security layer callable from the connection layer rather than the application and wherein the security layer is unbeknownst to the application." Independent claim 16 includes limitations for "at a security layer, receiving a request from a lower connection layer of a transport stack to establish a secure connection, which the application that utilizes the security layer is unaware of the security layer and its operations" and "the application directly receives the decrypted application data without being required to perform security procedure handshakes for secure network communications and without being aware of security communications that occur via the security layer." These limitations are not taught in the Tumblin reference as asserted by the Examiner and in fact are taught away from in the Tumblin reference because the Tumblin reference has taken a different approach entirely from what is recited within independents claim 1, 7, and 16.

The thrust of the Examiner's argument is a single passage in Tumblin at column 8 lines 22-25 and in FIG. 12 reference numeral 710 where the NSIM "intercepts" requests from the security non-extensible application. However, a single usage of the term "intercept" and a single passage taken out of context of the entire specification and teaching of Tumblin is not permissible.

Specifically, the Board's attention is directed to FIGS. 7-9 and related specification in Tumblin. Here, it can be clearly seen that the security non-extensible application includes an API to interact with the NSIM. The NSIM is a network security interface module or the module supplying security in the Tumblin reference. Tumblin unambiguously states that "[e]ach client NSIM 290 provides the network API 190 recognized by security non-extensible application 210." *Emphasis added.* Tumblin, column 5 lines 21-23. It is also stated that "[e]ach NSIM . . . is capable of receiving requests fro network services from the client program 210 . . . to which it is linked." The client program 210 is the security non-extensible application. The NSIM is linked to the security non-extensible application and provides an API to the security non-extensible application. See also column 8 lines 12-18 where it is stated that the security non-

extensible applications are linked to the NSIM's via the API. Column 8 lines 19-28 details how it is that this occurs a module in the security non-extensible application has been substituted with the NSIM. In other words, the application 210 (security non-extensible program) was made to interface with the security module by substituting its original access module with the NSIM and linking it and the API that supports it into the executable. This is not as the Examiner suggests an application that is security unaware or "unbeknownst to the application."

In fact, the application 210 is aware of the security because the application is linked to the NSIM and the API. To illustrate this point further, the Board's attention is directed to the following references within Tumblin: column 5 lines 14-24, a security non extensible program is linked to an API to replace that program's existing API to provide enhanced security; column 6 lines 19-24, the security non extensible program is linked to the API providing enhanced security; column 5 lines 34-41, the NSIM and the security non extensible are linked together via the API; column 5 lines 41-51 each NSIM is capable of interacting with the security non extensible program or application to which it is linked; column 8 lines 10 through 17 even specifically states the security non extensible programs are "linked" to the NSIM via an API; and claim 1 states that the NSIM is linked to the application program and so does claim 4.

In fact, the figures and the claims clearly state that the mechanism by which Tumblin achieves interaction or interception is to link into the application and API that calls the NSIM. The fact that Tumblin in passing uses "intercept" is inconsequential semantics that when the whole of Tumblin is read becomes clear and can be put in the proper context, which is from the source's point of view it may appear to be intercepted but from the executable's point of view a very specific mechanism is used to interface the security non-extensible module and the NSIM and that is to substitute in an API that directly communicates with the NSIM. So, the executable actually and clearly and directly communicates with the NSIM via a "linked" in API call.

This approach in Tumblin is different from what Applicants have taught. There is no modification needed in Applicants' approach to the program (executing version); whereas in Tumblin an API has to be linked into the program or else Tumblin cannot work. Thus, Applicants continue to maintain that Tumblin is security aware and is not security unaware. This is based on the fact that Tumblin is using a different arrangement and achieving its results in an

entirely different manner where the program being extended is actually linked to an API of the NSIM.

Tumblin has not taught an application 210 that is capable of passing application data to a security layer "unbeknownst to the application." The application 210 is Tumblin is aware and does know that it is passing data to the NSIM – it has to know because it directly passes the data to the NSIM via a linked in API. This is not what is done in the independent claims, where an application "directly" sends application data to an upper connection layer and not directly to a security layer. The two approaches are different from one another.

The crux of the issue is whether Tumblin can be said to teach a security unaware application. Appellants assert that this cannot be said because it is clear from the teachings in Tumblin that the application 210 is linked and interfaced directly to the NSIM and its API; thus, the application 210 cannot be said to be unaware of the security supplied by the NSIM.

This is not an insignificant point because the Tumblin reference is not capable of providing security to legacy application not having security features without first recompiling or at least re-linking such an application to specifically be aware of and call an NSIM. The applications in Tumblin must directly call the security module or NSIM, they do not indirectly interact with an upper connection layer that then handles connecting a security layer as Appellant has claimed. Appellant can supply security to legacy applications without those applications even having any awareness whatsoever because the interface is achieved in the protocol stack and not via the application as Tumblin elected to do. In other words, Tumblin achieves security integration by modifying the interface of the application to specifically call a security module (NSIM); Appellant achieves security integration by modifying the protocol stack independent entirely from the application's interface and calling mechanisms. This is a substantial distinguishing feature from Tumblin and one that cannot be glossed over because of a single use of a term that was taken entirely out of context, namely the use of "intercept."

Accordingly, Appellant believes the rejections do not provide a prima facie showing of anticipation and should be withdrawn and the claims allowed. Appellant respectfully requests the same of the Board.

D) Discussion of the rejections of claims 3 and 10 under 35 U.S.C. § 103(a) as being unpatentable over Tumblin et al. in view of SSL-Talk List FAQ.

Claim 3 is dependent from independent claim 1 and claim 10 is dependent from independent claim 7. Consequently, Appellant asserts that these claims are allowable in view of the remarks presented above with respect to the independent claims. Thus, Appellant respectfully requests an indication that claims 3 and 10 are allowable from the Board.

E) Discussion of the rejection of claim 5 under 35 USC § 103(a) as being unpatentable over Tumblin in view of Samar.

Claim 5 is dependent from independent claim 1. Therefore, claim 5 should be allowable in view of the remarks presented above with respect to independent claim 1. Appellant respectfully requests an indication of the same.

F) Discussion of the rejection of claims 11 and 19 under 35 USC § 103(a) as being unpatentable over Tumblin in view of What's Enhanced in NetWare 5.

Claim 11 is dependent from independent claim 7 and claim 19 is dependent from independent claim 16. Accordingly, Appellant asserts that claims 11 and 19 are allowable in view of the remarks presented above for claims 7 and 16 and respectfully requests an indication of the same.

G) Discussion of the rejection of claim 13 under 35 USC § 103(a) as being unpatentable over Tumblin in view of MS SSL Advisor.

Claim 13 is dependent from independent claim 7. Therefore, claim 13 should be allowable in view of the remarks presented above with respect to claim 7. Accordingly, Appellant respectfully requests an indication from the Board that claim 13 is allowable.

8. SUMMARY

For the reasons argued above, the independent claims were not properly rejected under § 102(e) as being anticipated by Tumblin.

It is respectfully submitted that the art cited does not render the claims anticipated or obvious and that the claims are patentable over the cited art. Reversal of the rejections and allowance of the pending claims are respectfully requested.

Respectfully submitted,

BABER AMIN et al.

By their Representatives,

SCHWEGMAN, LUNDBERG, WOESSNER & KLUTH, P.A.
P.O. Box 2938
Minneapolis, MN 55402

Date 03/06/07 By /Joseph P. Mehrle/
Joseph P. Mehrle
Reg. No. 45,535

CERTIFICATE UNDER 37 CFR 1.8: The undersigned hereby certifies that this correspondence is being deposited with the United States Postal Service with sufficient postage as first class mail, in an envelope addressed to: Mail Stop Appeal Brief, Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450, on this 6 day of March 2007.

Name Peter Rebuffoni Signature

CLAIMS APPENDIX

1. A method of providing transport-independent secure communications in a computer network, comprising the steps of:

directly receiving application data, from an application, at an upper connection layer of a transport protocol stack, wherein the application data is received from the application using a connection specific application programming interface (API) desired for communication by the application and which is not associated with security;

passing the application data from the upper connection layer to a security layer from within the transport protocol stack and unbeknownst to the application;

encrypting the application data within the security layer;

passing the encrypted application data from the security layer to a lower connection layer of the transport protocol stack; and

sending the encrypted application data from the lower connection layer out a network connection;

wherein the application is not required to perform security handshakes in order to send encrypted application data over the network, the connection layers support at least one network transport protocol, and the security layer is not specific to that transport protocol.

2. The method of claim 1, further comprising the steps of receiving at the lower connection layer encrypted application data which came in at the network connection; passing the encrypted application data from the lower connection layer to the security layer; decrypting the application data within the security layer; passing the decrypted application data from the security layer to the upper connection layer; and sending the decrypted application data from the upper connection layer to the application, without requiring that the application perform a security handshake.

3. The method of claim 1, further comprising the step of the lower connection layer establishing a connection with a handshake mode that is at least one of an interactive mode and a blind-root-accept mode.

4. The method of claim 1, further comprising the step of the lower connection layer establishing a connection with a handshake mode that is at least one of a server mode, a client mode, and a server with client authentication enabled mode.

5. The method of claim 1, further comprising the step of changing a list of trusted roots for the secure connection.

6. The method of claim 1, further comprising the step of the security layer informing at least one of the connection layers of security handshake proceedings.

7. A system for secure computer networking, comprising:

an application which is free of code for performing security procedure handshakes for secure network communications;

at least one connection layer directly interfaced with the application, the connection layer comprising an upper connection layer associated with a transport protocol stack and a lower connection layer associated with the transport protocol stack, the connection layers comprising code for performing at least one network transport protocol; and

a security layer callable from the connection layer rather than the application and wherein the security layer is unbeknownst to the application, the security layer comprising code for performing security procedure handshakes for secure network communications, the security layer also comprising code for encrypting and decrypting application data, and wherein the application initially sends application data to the protocol stack of the upper connection layer directly using a desired application programming interface (API) associated with a connection mechanism that is not associated with security.

8. The system of claim 7, wherein the connection layers comprise code for performing a WinSock network transport protocol.

9. The system of claim 7, wherein the security layer comprises code for performing security procedure handshakes for a Secure Sockets Layer session.

10. The system of claim 7, wherein the security layer comprises code for performing security procedure handshakes for a Transport Layer Security session.

11. The system of claim 7, wherein the application comprises code for providing Lightweight Directory Access Protocol services.

12. The system of claim 7, comprising a means for the security layer and at least one of the connection layers to identify a particular application and its cryptographic properties.

13. The system of claim 7, comprising a means for the security layer and at least one of the connection layers to identify a function as a call back function.

14. The system of claim 7, comprising a means for establishing a secure connection using a specified handshake mode.

15. The system of claim 7, further comprising a legacy application which performs security handshakes, and a security module supporting a secure connection to the legacy application.

16. A configured storage medium embodying data and instructions readable by a computer to perform a method of processing application data for secure network communications, the method comprising the computer-implemented steps of:

at a security layer, receiving a request from a lower connection layer of a transport protocol stack to establish a secure connection, wherein an application that utilizes the security layer is unaware of the security layer and its operations;

in response, utilizing a means for establishing a connection to establish the requested connection; and

at the security layer, receiving encrypted application data from the lower connection layer, decrypting the application data, and passing the decrypted application data to an upper connection layer of the transport protocol stack;

whereby the application directly receives the decrypted application data without being required to perform security procedure handshakes for secure network communications and without being aware of security communications that occur via the security layer, and wherein the application receives the decrypted application data in a desired application programming interface (API) associated with a connection that the application originally used and that is not associated with security.

17. The configured storage medium of claim 16, wherein the means for establishing a connection establishes a Secure Sockets Layer connection.

18. The configured storage medium of claim 16, wherein the method further comprises receiving the encrypted application data at the lower connection layer using a transport model.

19. The configured storage medium of claim 18, wherein the lower connection layer uses a proxy transport model.

20. The configured storage medium of claim 16, further comprising a signal embodied in the computer, the signal comprising a secure network communications protocol stack interface which is callable from at least the lower connection layer.

EVIDENCE APPENDIX

None.

RELATED PROCEEDINGS APPENDIX

None.